

DM 2 Lab Report

2023-06-07

Initial summaries and plots - EDA

We will begin by loading our data sets in R:

```
attributes<- read.csv('attributes.csv')
edges<-read.csv('edges.csv')
head(attributes)
```

```
##   name   Age Gender SleepLoc
## 1     1 young   Male     Loc2
## 2     2 young   Male     Loc2
## 3     3 young   Male     Loc2
## 4     4 young   Male     Loc1
## 5     5 young   Male     Loc1
## 6     6 young   Male     Loc2
```

```
head(edges)
```

```
##   V1 V2
## 1  1  7
## 2  1  8
## 3  1 11
## 4  1 12
## 5  1 14
## 6  1 15
```

As we can see, the attributes data set gives us information about each of the nodes (these nodes are indexed), such as the chimpanzees age, gender and preferred sleeping location. The edges data set lists for each node, all the nodes it is connected to. Since it is a directed graph, the connection is from V1 to V2.

We will now set up the network of our matrix:

```
matrix<- as.matrix(edges)
library(statnet)

## Loading required package: tergm
## Loading required package: ergm
## Loading required package: network

##
## 'network' 1.18.1 (2023-01-24), part of the Statnet Project
## * 'news(package="network")' for changes since last version
```

```

## * 'citation("network")' for citation information
## * 'https://statnet.org' for help, support, and other information

##
## 'ergm' 4.4.0 (2023-01-26), part of the Statnet Project
## * 'news(package="ergm")' for changes since last version
## * 'citation("ergm")' for citation information
## * 'https://statnet.org' for help, support, and other information

## 'ergm' 4 is a major update that introduces some backwards-incompatible
## changes. Please type 'news(package="ergm")' for a list of major
## changes.

## Loading required package: networkDynamic

##
## 'networkDynamic' 0.11.3 (2023-02-15), part of the Statnet Project
## * 'news(package="networkDynamic")' for changes since last version
## * 'citation("networkDynamic")' for citation information
## * 'https://statnet.org' for help, support, and other information

## Registered S3 method overwritten by 'tergm':
##   method                from
##   simulate_formula.network ergm

##
## 'tergm' 4.1.1 (2022-11-07), part of the Statnet Project
## * 'news(package="tergm")' for changes since last version
## * 'citation("tergm")' for citation information
## * 'https://statnet.org' for help, support, and other information

##
## Attaching package: 'tergm'

## The following object is masked from 'package:ergm':
##
##   snctrl

## Loading required package: ergm.count

##
## 'ergm.count' 4.1.1 (2022-05-24), part of the Statnet Project
## * 'news(package="ergm.count")' for changes since last version
## * 'citation("ergm.count")' for citation information
## * 'https://statnet.org' for help, support, and other information

## Loading required package: sna

## Loading required package: statnet.common

##
## Attaching package: 'statnet.common'

```

```

## The following object is masked from 'package:ergm':
##
##      snctrl

## The following objects are masked from 'package:base':
##
##      attr, order

## sna: Tools for Social Network Analysis
## Version 2.7-1 created on 2023-01-24.
## copyright (c) 2005, Carter T. Butts, University of California-Irvine
## For citation information, type citation("sna").
## Type help(package="sna") to get started.

## Loading required package: tsna

##
## 'statnet' 2019.6 (2019-06-13), part of the Statnet Project
## * 'news(package="statnet")' for changes since last version
## * 'citation("statnet")' for citation information
## * 'https://statnet.org' for help, support, and other information

## unable to reach CRAN

net1<- network(matrix, matrix.type='edgelist')
#summary(network)
#add gender information
network::set.vertex.attribute(net1, "Gender",attributes$Gender)
network::set.vertex.attribute(net1, "SleepLoc",attributes$SleepLoc)
network::set.vertex.attribute(net1, "Age",attributes$Age)

```

We will now visualize our network:

```

#convert to igraph
library(intergraph)
library(igraph)

##
## Attaching package: 'igraph'

## The following objects are masked from 'package:sna':
##
##      betweenness, bonpow, closeness, components, degree, dyad.census,
##      evcent, hierarchy, is.connected, neighborhood, triad.census

## The following objects are masked from 'package:network':
##
##      %c%, %s%, add.edges, add.vertices, delete.edges, delete.vertices,
##      get.edge.attribute, get.edges, get.vertex.attribute, is.bipartite,
##      is.directed, list.edge.attributes, list.vertex.attributes,
##      set.edge.attribute, set.vertex.attribute

```

```

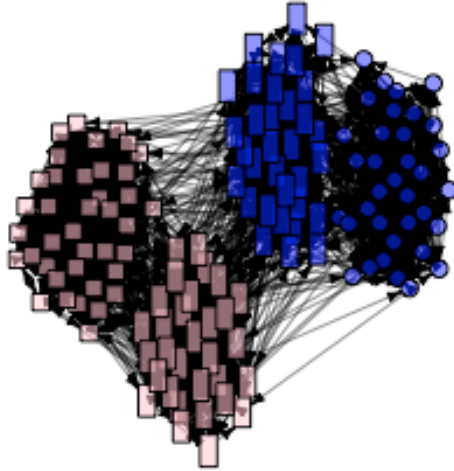
## The following objects are masked from 'package:stats':
##
##      decompose, spectrum

## The following object is masked from 'package:base':
##
##      union

inet1<-asIgraph(net1)
#colour according to gender
transp<- function(col, alpha=0.5){
  res<- apply(col2rgb(col), 2, function(c) rgb(c[1]/255,c[2]/255,c[3]/255,
alpha))
  return(res)
}
Fcolor<- transp('pink', alpha=0.5)
Mcolor<- transp('blue', alpha=0.5)
V(inet1)$color<- V(inet1)$Gender
V(inet1)$color<- gsub('Female', Fcolor, V(inet1)$color)
V(inet1)$color<- gsub('Male', Mcolor, V(inet1)$color)
#shape according to age
V(inet1)$shape<- NA
V(inet1)$shape<- ifelse(V(inet1)$Age=='young', 'circle', 'tomatoe')
V(inet1)$shape<- ifelse(V(inet1)$Age=='elderly', 'square', V(inet1)$shape)
V(inet1)$shape<- ifelse(V(inet1)$Age=='middle-aged', 'rectangle',
V(inet1)$shape)

plot(inet1, vertex.size=8, arrowhead.cex=0.5, edge.lty=1,edge.width=0.5,
edge.color="black", edge.arrow.size=0.3, vertex.label=NA,
vertex.shape=V(inet1)$shape)

```



We have edited this visualization, so the color represents the gender (blue=male, pink=female) and the shape represents the age (circle=young, rectangle=middle-aged, square=elderly).

We can see that there are 4 main segregated communities. It seems as though genders and ages are entirely separated.

We will now perform explanatory data analysis on the attributes of the chimpanzees.

```
summary(attributes)
```

##	name	Age	Gender	SleepLoc
##	Min. : 1.00	Length:138	Length:138	Length:138
##	1st Qu.: 35.25	Class :character	Class :character	Class :character
##	Median : 69.50	Mode :character	Mode :character	Mode :character
##	Mean : 69.50			
##	3rd Qu.:103.75			
##	Max. :138.00			

```
library(ggplot2)
#library(cowplot)
library(gridExtra)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

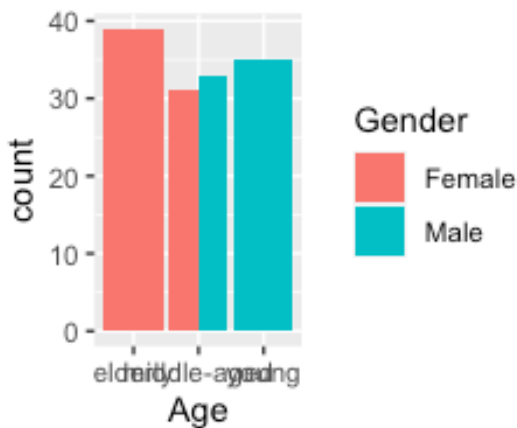
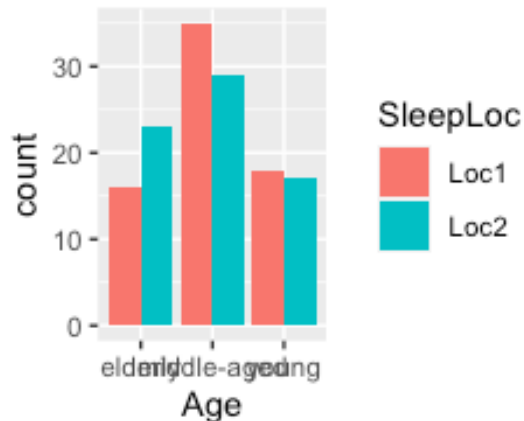
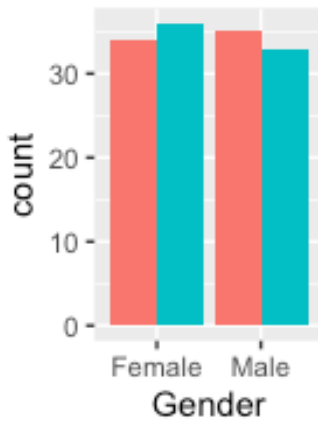
## The following object is masked from 'package:gridExtra':
##
##      combine

## The following objects are masked from 'package:igraph':
##
##      as_data_frame, groups, union

## The following objects are masked from 'package:stats':
##
##      filter, lag

## The following objects are masked from 'package:base':
##
##      intersect, setdiff, setequal, union

library(cowplot)
#library(dplyr)
one<- ggplot(attributes, aes(x=Gender, fill=SleepLoc)) +
  geom_bar(stat="count", position = "dodge")
two<-ggplot(attributes, aes(x=Age, fill=SleepLoc)) +
  geom_bar(stat="count", position = "dodge")
three<-ggplot(attributes, aes(x=Age, fill=Gender)) +
  geom_bar(stat="count", position = "dodge")
plot_grid(one, two, three,ncol = 2, nrow = 2)
```



There are 138 chimpanzees in total. From the Bar charts, it is shown that all the elderly chimpanzees are female, all the young are male and the middle aged are approximately evenly divided in gender. There is approximately an even number of Male and Female chimpanzees in total and an even spread in location 1 and 2. majority of the chimpanzees are middle aged.

```
prop.table(table(attributes$Age))

##
##      elderly middle-aged      young
##  0.2826087  0.4637681  0.2536232

prop.table(table(attributes$Gender))

##
##      Female      Male
##  0.5072464  0.4927536

prop.table(table(attributes$SleepLoc))

##
##   Loc1 Loc2
##   0.5  0.5
```

These proportions further summarize information. There is an even distribution of the gender and the location preferences. There is double the number of middle aged chimpanzees compared to the number of elderly and young.

Check for missing data:

```
attributes[!complete.cases(attributes), ]  
## [1] name      Age      Gender  SleepLoc  
## <0 rows> (or 0-length row.names)
```

No missing data.

We will now examine various measures of the network.

Density:

```
network.density(net1)  
## [1] 0.1420184
```

The density of a graph represents how many edges are observed versus how many edges are possible. In this graph, 0.14% of the possible number of connections are connected.

Reciprocity:

```
grecip(net1, measure = "dyadic.nonnull")  
##      Mut  
## 0.2339154
```

The reciprocity of a graph measures the proportion of times a tie is reciprocated (an edge connected on both edges). This means that 23.3% of the ties are reciprocated. This means that 23.3% of the times that a chimpanzee grooms another chimpanzee, the action is reciprocated.

Diameter:

```
max(geodist(net1)$gdist)  
## [1] 4
```

This means that in any case, it takes at maximum 4 steps to get from one chimpanzee to another.

Transitivity:

This is a measure of the tendency for three nodes to be fully connected (forming a triangle).

```
suppressWarnings(gtrans(net1, mode="weak"))  
## [1] 0.3165853
```



```
suppressWarnings(gtrans(net1, mode="strong"))  
## [1] 0.3165853
```

This means that 31.65% of the chimpanzees with at least two ties between them will have the third tie.

Cliques:

A clique is a sub graph which has a density of 1 - all nodes in that graph are maximum connected.

```
library(igraph)  
library(intergraph)  
igraph::clique.number(asIgraph(net1))  
  
## Warning in igraph::clique.number(asIgraph(net1)): At  
## core/cliques/maximal_cliues_template.h:269 : Edge directions are ignored  
## for  
## maximal clique calculation.  
  
## [1] 10  
  
igraph::cliues(asIgraph(net1))
```

We can create a sub graph where at most 10 chimpanzees are fully connected – all 10 chimpanzees are fully connected in this clique. The second command can be run to identify these cliques.

Node Degree:

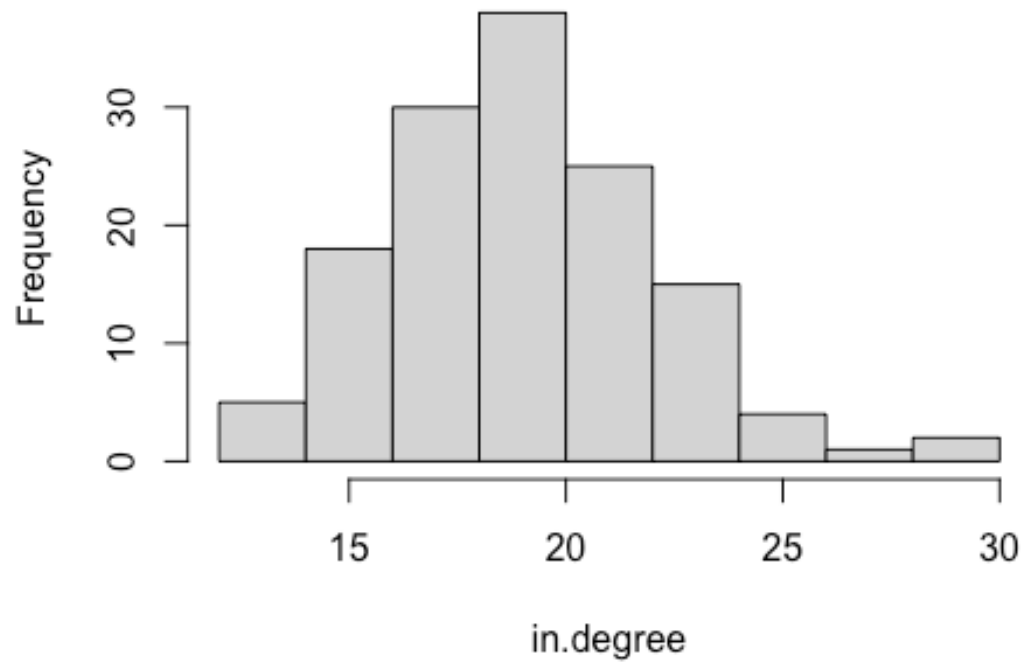
It is useful to assess if there are any nodes that hold particular importance on the network. In-degree is the number of edges leading into a node (groomed) and out-degree as the number of edges leading out of a node (grooming).

```
in.degree<-sna::degree(net1, cmode="indegree")  
out.degree<-sna::degree(net1, cmode="outdegree")
```

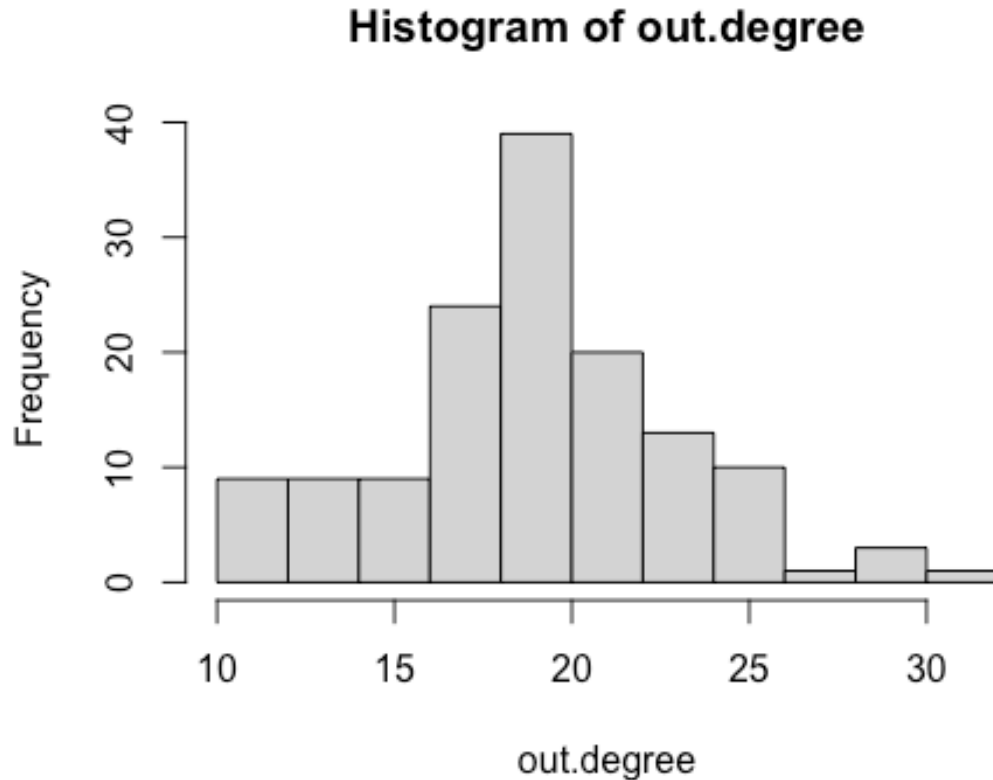
This lists the in-degree and out-degree of each chimpanzee. As there are many chimpanzees in this list, we can represent this information in a histogram.

```
hist(in.degree)
```

Histogram of in.degree



```
hist(out.degree)
```



The distributions of both histograms are normal, the mean in and out degree is 20. This is further supported by the summary statistics below.

```
summary(in.degree)
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    12.00  17.00   19.00   19.46  21.75   29.00

summary(out.degree)
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    10.00  17.00   20.00   19.46  21.75   31.00
```

It seems that every chimpanzee has at least 12 and at most 29 inward connections and at least 10 and at most 31 outward connections.

```
order(in.degree, decreasing = TRUE)[1:5]
## [1]  8 100 123  37  47

order(out.degree, decreasing = TRUE)[1:5]
## [1]  54  52 131 135 126
```

Chimpanzee 8 has the most inward connection - the most frequently groomed chimpanzee while chimpanzee 54 has the most outward connection - it grooms the most.

Centrality:

Centrality refers to a particular chimpanzee which holds importance in the network. We will first assess the closeness centrality. This measures how close each node is to all other nodes in the graph.

```
cent<- sna::closeness(net1, gmode = "digraph")
max(sna::closeness(net1, gmode = "digraph"))

## [1] 0.5393701

order(cent, decreasing = TRUE)[1:5]

## [1] 126 42 71 84 80
```

The large numbers represent a chimpanzee that is close to many other chimpanzees whereas small numbers represent chimpanzees that are far away from most other chimpanzees. We can see that chimpanzee 126 has a high closeness centrality of 0.539.

Betweenness centrality: Betweenness measures how important a node is in getting from one node to another node.

```
cent2<-sna::betweenness(net1, cmode="directed")
max(cent2)

## [1] 385.7442

order(cent2, decreasing = TRUE)[1:5]

## [1] 126 47 84 42 28
```

Once again, chimpanzee 126 has the highest betweenness centrality. This means that if 126 was removed from the network, chimpanzees will not have as much access to each other.

Assortativity:

We will now evaluate if there are trends that can be drawn between the characteristics of the chimpanzees to explain why certain chimpanzees are well connected. If a graph is very assortive, we expect that all pairs of nodes with similar characteristics have a high connection between them. When there is perfect assortative mixing, the index will be 1 and will be close to -1 when it is disassortative.

```
gend<- as.factor(attributes$Gender)
assortativity_nominal(asIgraph(net1), gend)

## [1] 0.8382416
```

This is very close to 1 which suggests that chimpanzees of similar genders are more likely to possess ties.

```
age<- as.factor(attributes$Age)
assortativity_nominal(asIgraph(net1), age)

## [1] 0.7458868

sleep<- as.factor(attributes$SleepLoc)
assortativity_nominal(asIgraph(net1), sleep)

## [1] -0.01464125
```

The high positive value for age suggest that chimpanzees of similar age are more likely to be connected. The close to zero value for sleeping location suggests that the connections are not connected to the preferred sleeping location of the chimpanzees. Chimpanzees will connect to those in the same and different preferred sleeping location

ERGM analysis

We will now attempt to understand the network statistically, using ERGM analysis.

In the null model, we are interested in the density of the graph which is essentially the number of edges.

```
library(ergm)
modell<-ergm(net1 ~ edges, control=control.ergm(seed=101))

## Starting maximum pseudolikelihood estimation (MPLE):
## Evaluating the predictor and response matrix.
## Maximizing the pseudolikelihood.
## Finished MPLE.
## Stopping at the initial estimate.
## Evaluating log-likelihood at the estimate.

summary(modell)

## Call:
## ergm(formula = net1 ~ edges, control = control.ergm(seed = 101))
##
## Maximum Likelihood Results:
##
##      Estimate Std. Error MCMC % z value Pr(>|z|)
## edges -1.79863    0.02083      0 -86.33  <1e-04 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      Null Deviance: 26209  on 18906  degrees of freedom
##      Residual Deviance: 15450  on 18905  degrees of freedom
##
## AIC: 15452  BIC: 15460  (Smaller is better. MC Std. Err. = 0)
```

```

plogis(coef(model1))

##      edges
## 0.1420184

#compare to the density
network.density(net1)

## [1] 0.1420184

#graph density
exp(-1.79863)/(1+exp(-1.79863))

## [1] 0.1420179

```

This model has produced the graph density. The probability of any tie existing in the entire network is 0.142.

Now we will assess the mutual model:

```

model1.01<- ergm(net1~mutual + edges)

## Starting maximum pseudolikelihood estimation (MPLE):
## Evaluating the predictor and response matrix.
## Maximizing the pseudolikelihood.
## Finished MPLE.
## Starting Monte Carlo maximum likelihood estimation (MCMLE):
## Iteration 1 of at most 60:
## Warning: 'glpk' selected as the solver, but package 'Rglpk' is not
available;
## falling back to 'lpSolveAPI'. This should be fine unless the sample size
and/or
## the number of parameters is very big.
## Optimizing with step length 1.0000.
## The log-likelihood improved by 0.0381.
## Convergence test p-value: 0.1615. Not converged with 99% confidence;
increasing sample size.
## Iteration 2 of at most 60:
## Optimizing with step length 1.0000.
## The log-likelihood improved by 0.0428.
## Convergence test p-value: 0.0150. Not converged with 99% confidence;
increasing sample size.
## Iteration 3 of at most 60:
## Optimizing with step length 1.0000.
## The log-likelihood improved by 0.0063.

```

```
## Convergence test p-value: 0.0038. Converged with 99% confidence.
## Finished MCMLE.
## Evaluating log-likelihood at the estimate. Fitting the dyad-independent
submodel...
## Bridging between the dyad-independent submodel and the full model...
## Setting up bridge sampling...
## Using 16 bridges: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 .
## Bridging finished.
## This model was fit using MCMC. To examine model diagnostics and check
## for degeneracy, use the mcmc.diagnostics() function.
```

```
summary(model1.01)
```

```
## Call:
## ergm(formula = net1 ~ mutual + edges)
##
## Monte Carlo Maximum Likelihood Results:
##
##      Estimate Std. Error MCMC % z value Pr(>|z|)
## mutual  1.67252    0.06983      0  23.95  <1e-04 ***
## edges  -2.16650    0.02864      0 -75.64  <1e-04 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      Null Deviance: 26209 on 18906 degrees of freedom
## Residual Deviance: 14879 on 18904 degrees of freedom
##
## AIC: 14883 BIC: 14899 (Smaller is better. MC Std. Err. = 3.223)
```

Here, the number of mutual (reciprocated) ties is compared to a generated random networks with the same density. The effect is positive and significant indicating there are more mutual links in our network than what one would expect from a random network of with the same number of edges. In the context of the chimpanzees, this means that more often than expected, chimpanzees will reciprocate if groomed by another chimpanzee.

Now we will look beyond the null model and look at the covariates. We will first assess the correlation between each of the variables and the the degree of the node.

```
model2<-
ergm(net1~edges+mutual+nodefactor("SleepLoc")+nodefactor("Gender")+nodefactor
("Age"),control=control.ergm(seed=404))

## Starting maximum pseudolikelihood estimation (MPLE):

## Evaluating the predictor and response matrix.

## Maximizing the pseudolikelihood.

## Finished MPLE.

## Starting Monte Carlo maximum likelihood estimation (MCMLE):
```

```

## Iteration 1 of at most 60:
## Optimizing with step length 1.0000.
## The log-likelihood improved by 0.0772.
## Convergence test p-value: 0.6985. Not converged with 99% confidence;
increasing sample size.
## Iteration 2 of at most 60:
## Optimizing with step length 1.0000.
## The log-likelihood improved by 0.0588.
## Convergence test p-value: 0.0092. Converged with 99% confidence.
## Finished MCMLE.
## Evaluating log-likelihood at the estimate. Fitting the dyad-independent
submodel...
## Bridging between the dyad-independent submodel and the full model...
## Setting up bridge sampling...
## Using 16 bridges: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 .
## Bridging finished.
## This model was fit using MCMC. To examine model diagnostics and check
## for degeneracy, use the mcmc.diagnostics() function.

summary(model2)

## Call:
## ergm(formula = net1 ~ edges + mutual + nodefactor("SleepLoc") +
##       nodefactor("Gender") + nodefactor("Age"), control = control.ergm(seed
= 404))
##
## Monte Carlo Maximum Likelihood Results:
##
##               Estimate Std. Error MCMC % z value Pr(>|z|)
## edges          -2.09050    0.06049    0 -34.561  <1e-04 ***
## mutual           1.67405    0.06910    0  24.226  <1e-04 ***
## nodefactor.SleepLoc.Loc2 -0.02122    0.02646    0  -0.802    0.423
## nodefactor.Gender.Male    0.01363    0.03841    0   0.355    0.723
## nodefactor.Age.middle-aged -0.03656    0.03723    0  -0.982    0.326
## nodefactor.Age.young     -0.07347    0.05434    0  -1.352    0.176
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      Null Deviance: 26209 on 18906 degrees of freedom
## Residual Deviance: 14873 on 18900 degrees of freedom
##
## AIC: 14885 BIC: 14932 (Smaller is better. MC Std. Err. = 2.186)

```

These coefficients are insignificant. Ignoring the significance, the negative sleeping location coefficient suggests that sleeping location 2 is less likely to form ties than the chimpanzees who prefer sleeping location 1. The positive male coefficient suggests that males are more likely to form ties than females (although this difference is very small). The negative

coefficients for middle aged and young chimpanzees suggest that elderly chimpanzees are more likely to form ties than the other ages. Again, this difference is very small.

We will now assess the ties formed between different combinations in our network.

```
mixingmatrix(net1, 'Gender')
```

```
##           To
## From      Female Male  Sum
## Female    1272  119 1391
## Male       98 1196 1294
## Sum       1370 1315 2685
```

```
mixingmatrix(net1, 'Age')
```

```
##           To
## From      elderly middle-aged young  Sum
## elderly         662       121   17  800
## middle-aged      91      1039  101 1231
## young           12        95   547  654
## Sum             765      1255  665 2685
```

```
mixingmatrix(net1, 'SleepLoc')
```

```
##           To
## From      Loc1 Loc2  Sum
## Loc1     676  683 1359
## Loc2     679  647 1326
## Sum     1355 1330 2685
```

From this output, we can see that there are ties with each combination within a class. Specifically, there are very few ties between young and old chimpanzees and many ties between middle aged chimpanzees. This is in line with the explanatory analysis we did above. There are also more ties of the same gender than of different.

```
model3<-
ergm(net1~edges+mutual+nodematch("SleepLoc")+nodematch("Gender")+nodematch("Age"),
control=control.ergm(seed=404))
```

```
## Starting maximum pseudolikelihood estimation (MPLE):
```

```
## Evaluating the predictor and response matrix.
```

```
## Maximizing the pseudolikelihood.
```

```
## Finished MPLE.
```

```
## Starting Monte Carlo maximum likelihood estimation (MCMLE):
```

```
## Iteration 1 of at most 60:
```

```
## Optimizing with step length 1.0000.
```

```
## The log-likelihood improved by 0.1343.
## Estimating equations are not within tolerance region.
## Iteration 2 of at most 60:
## Optimizing with step length 1.0000.
## The log-likelihood improved by 0.1074.
## Estimating equations are not within tolerance region.
## Iteration 3 of at most 60:
## Optimizing with step length 1.0000.
## The log-likelihood improved by 0.2060.
## Estimating equations are not within tolerance region.
## Iteration 4 of at most 60:
## Optimizing with step length 1.0000.
## The log-likelihood improved by 0.1886.
## Estimating equations are not within tolerance region.
## Estimating equations did not move closer to tolerance region more than 1
time(s) in 4 steps; increasing sample size.
## Iteration 5 of at most 60:
## Optimizing with step length 1.0000.
## The log-likelihood improved by 0.0299.
## Convergence test p-value: 0.0449. Not converged with 99% confidence;
increasing sample size.
## Iteration 6 of at most 60:
## Optimizing with step length 1.0000.
## The log-likelihood improved by 0.0343.
## Convergence test p-value: 0.1764. Not converged with 99% confidence;
increasing sample size.
## Iteration 7 of at most 60:
## Optimizing with step length 1.0000.
## The log-likelihood improved by 0.0431.
## Convergence test p-value: 0.0146. Not converged with 99% confidence;
increasing sample size.
## Iteration 8 of at most 60:
## Optimizing with step length 1.0000.
## The log-likelihood improved by 0.0419.
## Convergence test p-value: 0.0141. Not converged with 99% confidence;
increasing sample size.
```

```

## Iteration 9 of at most 60:
## Optimizing with step length 1.0000.
## The log-likelihood improved by 0.0260.
## Convergence test p-value: < 0.0001. Converged with 99% confidence.
## Finished MCMLE.
## Evaluating log-likelihood at the estimate. Fitting the dyad-independent
submodel...
## Bridging between the dyad-independent submodel and the full model...
## Setting up bridge sampling...
## Using 16 bridges: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 .
## Bridging finished.
## This model was fit using MCMC. To examine model diagnostics and check
## for degeneracy, use the mcmc.diagnostics() function.

summary(model3)

## Call:
## ergm(formula = net1 ~ edges + mutual + nodematch("SleepLoc") +
##       nodematch("Gender") + nodematch("Age"), control = control.ergm(seed =
404))
##
## Monte Carlo Maximum Likelihood Results:
##
##               Estimate Std. Error MCMC % z value Pr(>|z|)
## edges          -4.74009    0.08451    0 -56.091   <1e-04 ***
## mutual           0.10304    0.07522    0   1.370    0.171
## nodematch.SleepLoc -0.01025    0.05095    0  -0.201    0.840
## nodematch.Gender   2.27655    0.07619    0  29.881   <1e-04 ***
## nodematch.Age      2.21872    0.06083    0  36.477   <1e-04 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      Null Deviance: 26209 on 18906 degrees of freedom
## Residual Deviance: 10843 on 18901 degrees of freedom
##
## AIC: 10853 BIC: 10893 (Smaller is better. MC Std. Err. = 0.1137)

```

Examining the full model with all variables included, mutual and sleeping location is insignificant. The coefficients of gender and age are positive which suggest that chimpanzees are more likely to have a tie if their age and gender match.

We will now drop the non significant term and create a new model:

```

model4<-
ergm(net1~edges+nodematch("Gender")+nodematch("Age"),control=control.ergm(see
d=404))

## Starting maximum pseudolikelihood estimation (MPLE):
## Evaluating the predictor and response matrix.

```

```

## Maximizing the pseudolikelihood.
## Finished MPLE.
## Stopping at the initial estimate.
## Evaluating log-likelihood at the estimate.
summary(model4)

## Call:
## ergm(formula = net1 ~ edges + nodematch("Gender") + nodematch("Age"),
##       control = control.ergm(seed = 404))
##
## Maximum Likelihood Results:
##
##              Estimate Std. Error MCMC % z value Pr(>|z|)
## edges           -4.76338    0.08025      0  -59.35  <1e-04 ***
## nodematch.Gender  2.30802    0.07461      0   30.93  <1e-04 ***
## nodematch.Age     2.25309    0.05693      0   39.58  <1e-04 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      Null Deviance: 26209  on 18906  degrees of freedom
## Residual Deviance: 10845  on 18903  degrees of freedom
##
## AIC: 10851  BIC: 10875  (Smaller is better. MC Std. Err. = 0)

```

The good news is that our terms are still significant and positive.

As the terms are significant, we can investigate to see if there is evidence of differential homophily at different levels of the factors.

```

model4.1<-
ergm(net1~edges+nodematch("Gender",diff=TRUE)+nodematch("Age",diff=TRUE),
      control=control.ergm(seed=404))

## Starting maximum pseudolikelihood estimation (MPLE):
## Evaluating the predictor and response matrix.
## Maximizing the pseudolikelihood.
## Finished MPLE.
## Stopping at the initial estimate.
## Evaluating log-likelihood at the estimate.
summary(model4.1)

## Call:
## ergm(formula = net1 ~ edges + nodematch("Gender", diff = TRUE) +

```

```
##      nodematch("Age", diff = TRUE), control = control.ergm(seed = 404))
##
## Maximum Likelihood Results:
##
##              Estimate Std. Error MCMC % z value Pr(>|z|)
## edges          -4.75504    0.08260      0 -57.57  <1e-04 ***
## nodematch.Gender.Female  2.30829    0.08600      0  26.84  <1e-04 ***
## nodematch.Gender.Male    2.28744    0.08511      0  26.88  <1e-04 ***
## nodematch.Age.elderly    2.23271    0.08052      0  27.73  <1e-04 ***
## nodematch.Age.middle-aged 2.24105    0.06382      0  35.11  <1e-04 ***
## nodematch.Age.young      2.30590    0.08510      0  27.10  <1e-04 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      Null Deviance: 26209 on 18906 degrees of freedom
## Residual Deviance: 10844 on 18900 degrees of freedom
##
## AIC: 10856 BIC: 10904 (Smaller is better. MC Std. Err. = 0)
```

The coefficients are positive and significant so we can conclude that chimpanzees are likely to have a connection between nodes of the same age and gender groups.

To test the ties between the different combinations of categories:

```
model4.2<-ergm(net1~edges+nodemix("Gender",base=1)+nodemix("Age",base=1),
control=control.ergm(seed=404))

## In term 'nodemix' in package 'ergm': Argument 'base' has been superseded
## by 'levels2', and it is recommended to use the latter. Note that its
## interpretation may be different.
## Starting maximum pseudolikelihood estimation (MPLE):
##
## Evaluating the predictor and response matrix.
##
## Maximizing the pseudolikelihood.
##
## Finished MPLE.
##
## Stopping at the initial estimate.
##
## Evaluating log-likelihood at the estimate.

summary(model4.2)

## Call:
## ergm(formula = net1 ~ edges + nodemix("Gender", base = 1) + nodemix("Age",
##      base = 1), control = control.ergm(seed = 404))
##
## Maximum Likelihood Results:
##
##              Estimate Std. Error MCMC % z value
```

```

Pr(>|z|)
## edges -0.21404 0.05225 0 -4.096 <1e-
04 ***
## mix.Gender.Male.Female -2.41859 0.12613 0 -19.176 <1e-
04 ***
## mix.Gender.Female.Male -2.27651 0.11785 0 -19.317 <1e-
04 ***
## mix.Gender.Male.Male -0.02048 0.08736 0 -0.234
0.815
## mix.Age.middle-aged.elderly -2.39814 0.12085 0 -19.844 <1e-
04 ***
## mix.Age.young.elderly -2.09255 0.32048 0 -6.529 <1e-
04 ***
## mix.Age.elderly.middle-aged -2.10626 0.10911 0 -19.304 <1e-
04 ***
## mix.Age.middle-aged.middle-aged 0.01642 0.08234 0 0.199
0.842
## mix.Age.young.middle-aged -2.27282 0.14542 0 -15.630 <1e-
04 ***
## mix.Age.elderly.young -1.88261 0.27602 0 -6.821 <1e-
04 ***
## mix.Age.middle-aged.young -2.22052 0.14286 0 -15.543 <1e-
04 ***
## mix.Age.young.young 0.07282 0.11724 0 0.621
0.535
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Null Deviance: 26209 on 18906 degrees of freedom
## Residual Deviance: 10836 on 18894 degrees of freedom
##
## AIC: 10860 BIC: 10955 (Smaller is better. MC Std. Err. = 0)

```

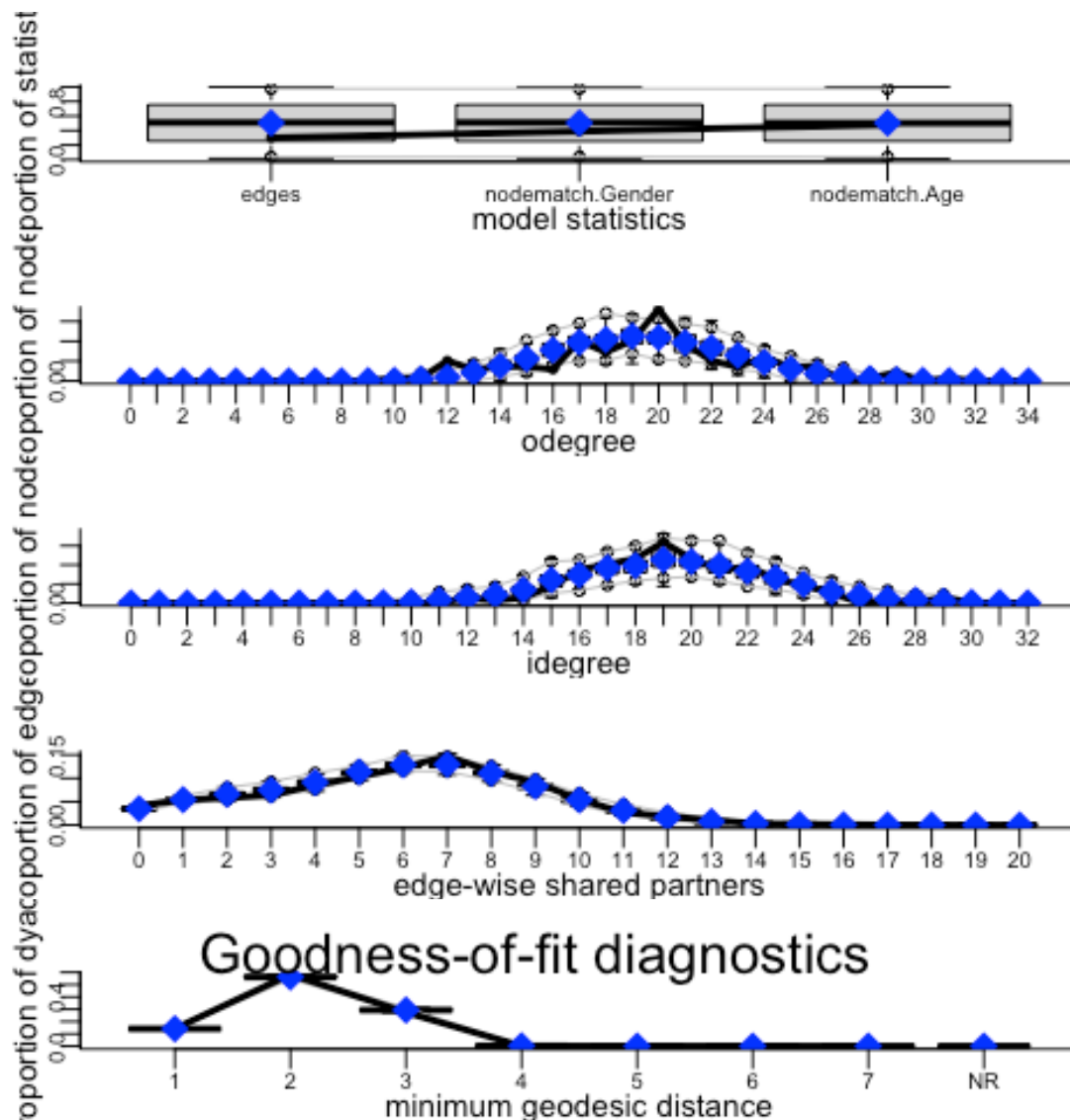
All the terms are significant (aside for the term comparing that of the same category). From the significant coefficients, we can see that opposite genders are less likely to attract and opposite ages are also less likely to attract.

Now that we established the model, we can check the fit of the model:

```

gof.model4<-gof(model4)
par(mfrow=c(4,1),cex=0.85, mgp=c(0.95,0.2,0), mai=c(0.325,0.45,0.325,0.05),
bty="L")
plot(gof.model4)

```



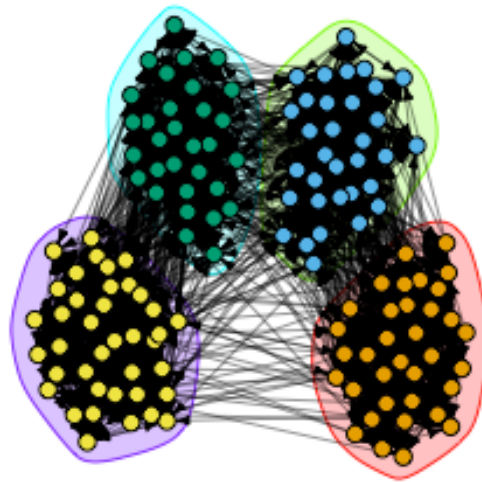
The minimum geodesic distance and edgewise shared partner plots show the observed line roughly in the middle of all box plots which is good. The model statistic line does deviate from the box plots a little. There is also some deviation in the in degree and out degree plots. Overall, this can be considered a good fit.

In conclusion, we have discovered that the chimpanzees are likely to be connected with those of the same gender and the same age group. We were not able to make any inference between the preferred sleeping location. This suggests that these chimpanzees are likely to find same-sex grooming partners.

Community detection and Stochastic block model (SBM)

1. Modularity based community detection

```
#convert to igraph
library(intergraph)
inet1<-asIgraph(net1)
# Perform Community detection
cs <- igraph::cluster_edge_betweenness(inet1)
plot(cs, inet1, vertex.size=8, arrowhead.cex=0.5, edge.lty=1, edge.width=0.5,
edge.color="black", edge.arrow.size=0.3, vertex.label=NA,
vertex.shape=V(inet1)$shape)
```

[illegible]


```
## 20 20 young Male Loc1 1
## 21 21 young Male Loc1 1
## 22 22 young Male Loc2 1
## 23 23 young Male Loc1 1
## 24 24 young Male Loc1 1
## 25 25 young Male Loc1 1
## 26 26 young Male Loc1 1
## 27 27 young Male Loc1 1
## 28 28 young Male Loc2 1
## 29 29 young Male Loc2 1
## 30 30 young Male Loc1 1
## 31 31 young Male Loc2 1
## 32 32 young Male Loc2 1
## 33 33 young Male Loc1 1
## 34 34 young Male Loc1 1
## 35 35 young Male Loc1 1
```

```
data%>%filter(cs$membership==2)
```

```
##      name      Age Gender SleepLoc cs$membership
## 1    36 middle-aged Male   Loc2         2
## 2    37 middle-aged Male   Loc2         2
## 3    38 middle-aged Male   Loc1         2
## 4    39 middle-aged Male   Loc2         2
## 5    40 middle-aged Male   Loc2         2
## 6    41 middle-aged Male   Loc2         2
## 7    42 middle-aged Male   Loc2         2
## 8    43 middle-aged Male   Loc1         2
## 9    44 middle-aged Male   Loc1         2
## 10   45 middle-aged Male   Loc1         2
## 11   46 middle-aged Male   Loc1         2
## 12   47 middle-aged Male   Loc1         2
## 13   48 middle-aged Male   Loc2         2
## 14   49 middle-aged Male   Loc2         2
## 15   50 middle-aged Male   Loc2         2
## 16   51 middle-aged Male   Loc1         2
## 17   52 middle-aged Male   Loc2         2
## 18   53 middle-aged Male   Loc1         2
## 19   54 middle-aged Male   Loc1         2
## 20   55 middle-aged Male   Loc2         2
## 21   56 middle-aged Male   Loc1         2
## 22   57 middle-aged Male   Loc2         2
## 23   58 middle-aged Male   Loc1         2
## 24   59 middle-aged Male   Loc1         2
## 25   60 middle-aged Male   Loc1         2
## 26   61 middle-aged Male   Loc2         2
## 27   62 middle-aged Male   Loc1         2
## 28   63 middle-aged Male   Loc2         2
## 29   64 middle-aged Male   Loc1         2
## 30   65 middle-aged Male   Loc2         2
```

```
## 31 66 middle-aged Male Loc1 2
## 32 67 middle-aged Male Loc1 2
## 33 68 middle-aged Male Loc2 2
```

```
data%>%filter(cs$membership==3)
```

```
##      name      Age Gender SleepLoc cs$membership
## 1  69 middle-aged Female   Loc1         3
## 2  70 middle-aged Female   Loc1         3
## 3  71 middle-aged Female   Loc2         3
## 4  72 middle-aged Female   Loc2         3
## 5  73 middle-aged Female   Loc1         3
## 6  74 middle-aged Female   Loc2         3
## 7  75 middle-aged Female   Loc1         3
## 8  76 middle-aged Female   Loc1         3
## 9  77 middle-aged Female   Loc1         3
## 10 78 middle-aged Female   Loc1         3
## 11 79 middle-aged Female   Loc1         3
## 12 80 middle-aged Female   Loc1         3
## 13 81 middle-aged Female   Loc2         3
## 14 82 middle-aged Female   Loc1         3
## 15 83 middle-aged Female   Loc1         3
## 16 84 middle-aged Female   Loc1         3
## 17 85 middle-aged Female   Loc1         3
## 18 86 middle-aged Female   Loc2         3
## 19 87 middle-aged Female   Loc1         3
## 20 88 middle-aged Female   Loc2         3
## 21 89 middle-aged Female   Loc1         3
## 22 90 middle-aged Female   Loc2         3
## 23 91 middle-aged Female   Loc2         3
## 24 92 middle-aged Female   Loc2         3
## 25 93 middle-aged Female   Loc2         3
## 26 94 middle-aged Female   Loc2         3
## 27 95 middle-aged Female   Loc1         3
## 28 96 middle-aged Female   Loc1         3
## 29 97 middle-aged Female   Loc2         3
## 30 98 middle-aged Female   Loc1         3
## 31 99 middle-aged Female   Loc2         3
```

```
data%>%filter(cs$membership==4)
```

```
##      name      Age Gender SleepLoc cs$membership
## 1  100 elderly Female   Loc1         4
## 2  101 elderly Female   Loc2         4
## 3  102 elderly Female   Loc2         4
## 4  103 elderly Female   Loc1         4
## 5  104 elderly Female   Loc2         4
## 6  105 elderly Female   Loc2         4
## 7  106 elderly Female   Loc2         4
## 8  107 elderly Female   Loc2         4
## 9  108 elderly Female   Loc1         4
```

## 10	109	elderly	Female	Loc2	4
## 11	110	elderly	Female	Loc1	4
## 12	111	elderly	Female	Loc1	4
## 13	112	elderly	Female	Loc1	4
## 14	113	elderly	Female	Loc2	4
## 15	114	elderly	Female	Loc2	4
## 16	115	elderly	Female	Loc2	4
## 17	116	elderly	Female	Loc1	4
## 18	117	elderly	Female	Loc2	4
## 19	118	elderly	Female	Loc2	4
## 20	119	elderly	Female	Loc1	4
## 21	120	elderly	Female	Loc2	4
## 22	121	elderly	Female	Loc2	4
## 23	122	elderly	Female	Loc2	4
## 24	123	elderly	Female	Loc1	4
## 25	124	elderly	Female	Loc2	4
## 26	125	elderly	Female	Loc2	4
## 27	126	elderly	Female	Loc1	4
## 28	127	elderly	Female	Loc1	4
## 29	128	elderly	Female	Loc1	4
## 30	129	elderly	Female	Loc2	4
## 31	130	elderly	Female	Loc1	4
## 32	131	elderly	Female	Loc1	4
## 33	132	elderly	Female	Loc2	4
## 34	133	elderly	Female	Loc2	4
## 35	134	elderly	Female	Loc1	4
## 36	135	elderly	Female	Loc2	4
## 37	136	elderly	Female	Loc2	4
## 38	137	elderly	Female	Loc1	4
## 39	138	elderly	Female	Loc2	4

The first cluster consists only young males. The second cluster consists of only middle-aged males. The third cluster consists of only middle-aged females. The fourth cluster consists of only elderly females. For all of these groups, there are chimpanzees from both preferred sleeping location.

2. Stochastic block model (SBM).

#convert edge list graph to adjacency matrix

```
library(igraph)
```

```
graph.data.frame(edges)
```

```
## IGRAPH c6d398e DN-- 138 2685 --
```

```
## + attr: name (v/c)
```

```
## + edges from c6d398e (vertex names):
```

```
## [1] 1->7 1->8 1->11 1->12 1->14 1->15 1->16 1->20 1->22 1->25
```

```
## [11] 1->30 1->33 1->34 1->43 1->67 2->5 2->6 2->7 2->8 2->11
```

```
## [21] 2->12 2->13 2->14 2->16 2->18 2->22 2->23 2->26 2->27 2->28
```

```
## [31] 2->34 2->64 2->109 2->130 3->2 3->5 3->6 3->7 3->8 3->12
```

```
## [41] 3->13 3->14 3->15 3->16 3->21 3->23 3->28 3->40 3->41 3->43
```

```
## [51] 3->49 3->60 3->79 3->110 4->1 4->7 4->9 4->11 4->13 4->16
```

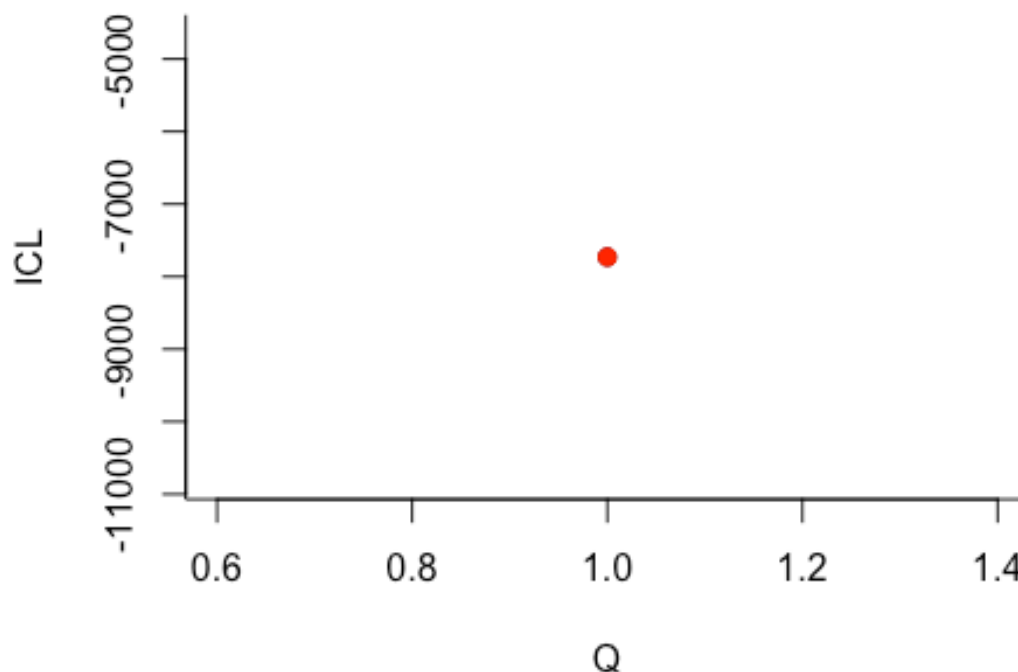
```
## [61] 4->17 4->22 4->24 4->25 4->26 4->27 4->28 4->29 4->30 4->31
## [71] 4->33 4->35 4->36 4->59 4->60 5->1 5->2 5->4 5->6 5->16
## + ... omitted several edges

edj.matrix<-get.adjacency(graph.data.frame(edges), sparse=FALSE)

library(blockmodels)
sbm.mod<-BM_bernoulli("SBM",as.matrix(edj.matrix))

sbm.mod$estimate()

## -> Estimation for 1 groups
##           -> 1 initializations provided
##           -> 0 initializations already used
##           -> Estimation with 1 initializations
## Executing 1 jobs in parallel
Better ICL criterion found
##           -> new ICL: -7730.11614800137
##           -> old ICL: NA
```

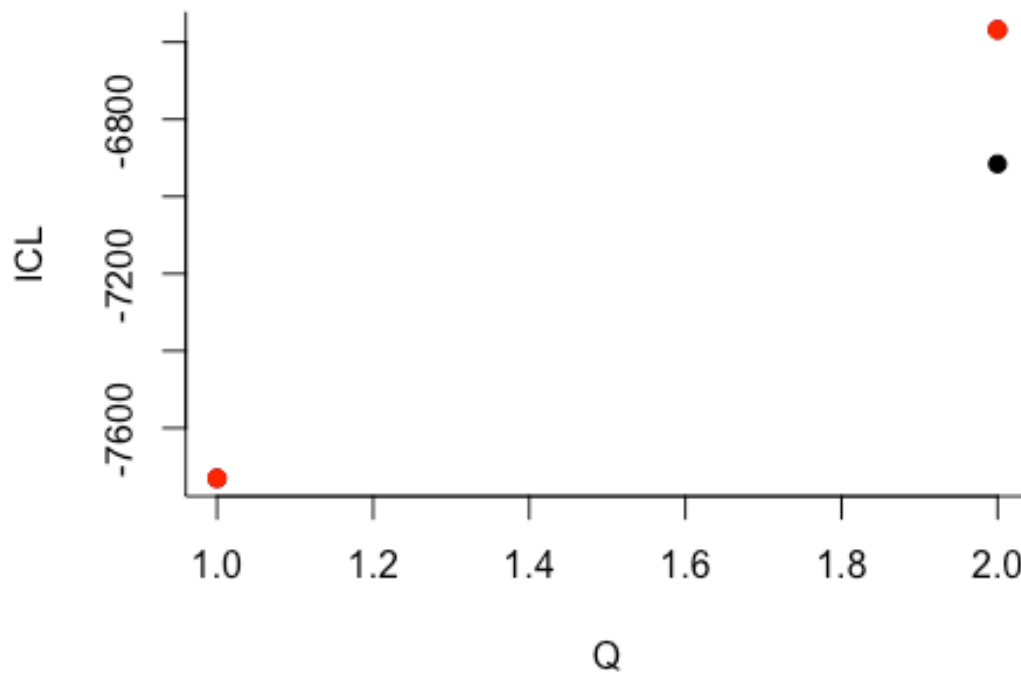


```
## -> Computation of eigen decomposition used for initializations
##
## -> Pass 1
##       -> With ascending number of groups
```

```

##      -> For 2 groups
##      -> Selecting initialization
##      -> Init from spectral clustering
##      -> Init from splitting groups from 1 groups
##      -> 2 initializations provided
##      -> 0 initializations already used
##      -> Estimation with 2 initializations
## Executing 2 jobs in parallel                                ->
Better ICL criterion found
##      -> new ICL: -6568.13973745724
##      -> old ICL: NA

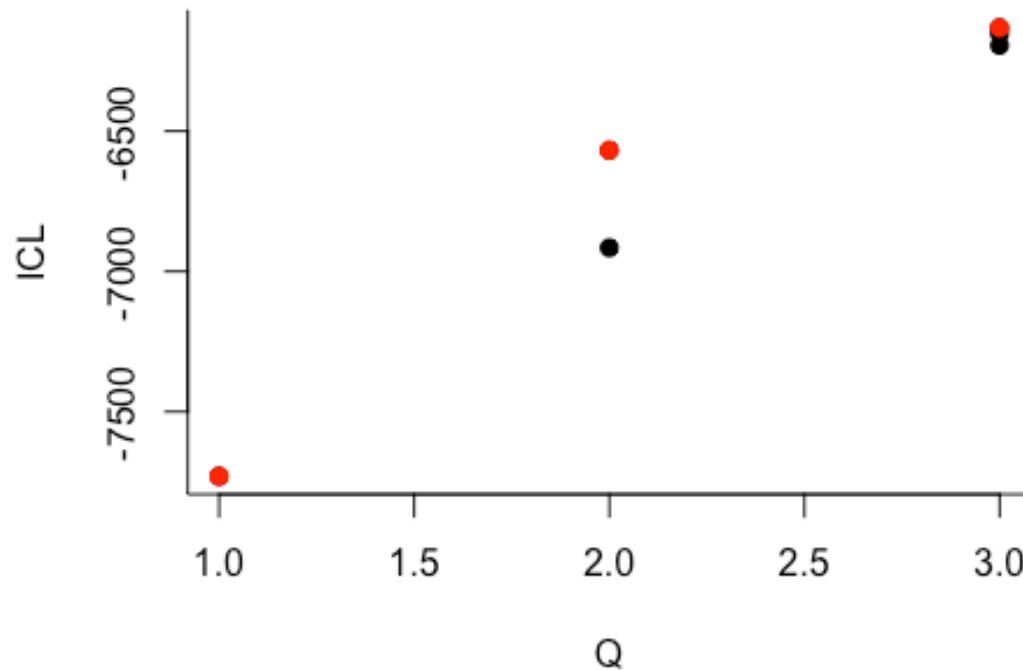
```



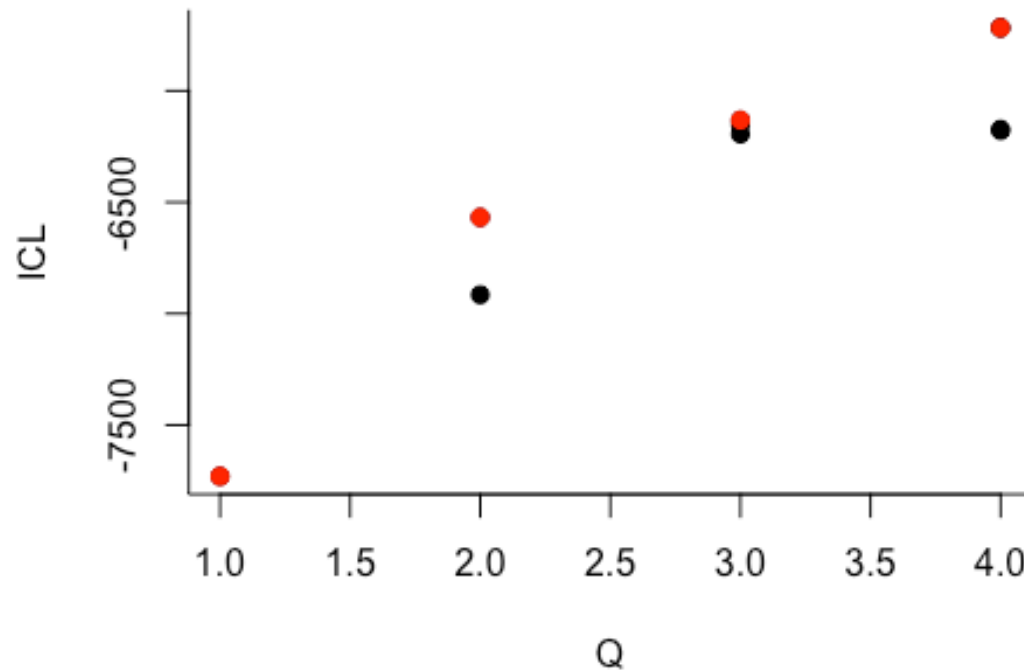
```

##      -> For 3 groups
##      -> Selecting initialization
##      -> Init from spectral clustering
##      -> Init from splitting groups from 2 groups
##      -> 3 initializations provided
##      -> 0 initializations already used
##      -> Estimation with 3 initializations
## Executing 3 jobs in parallel                                ->
Better ICL criterion found
##      -> new ICL: -6131.71308468876
##      -> old ICL: NA

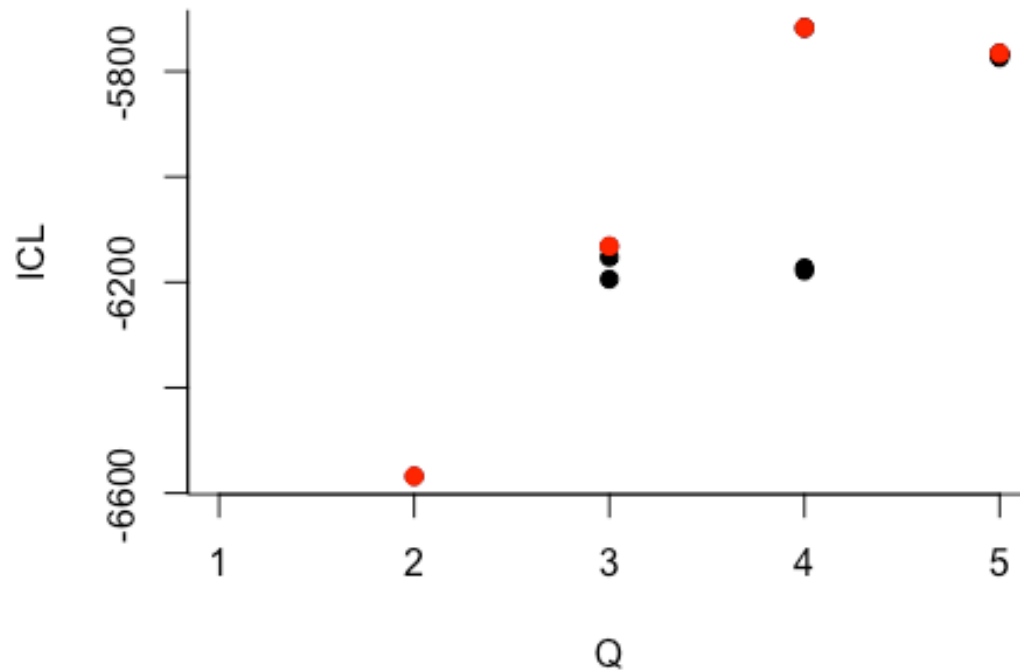
```



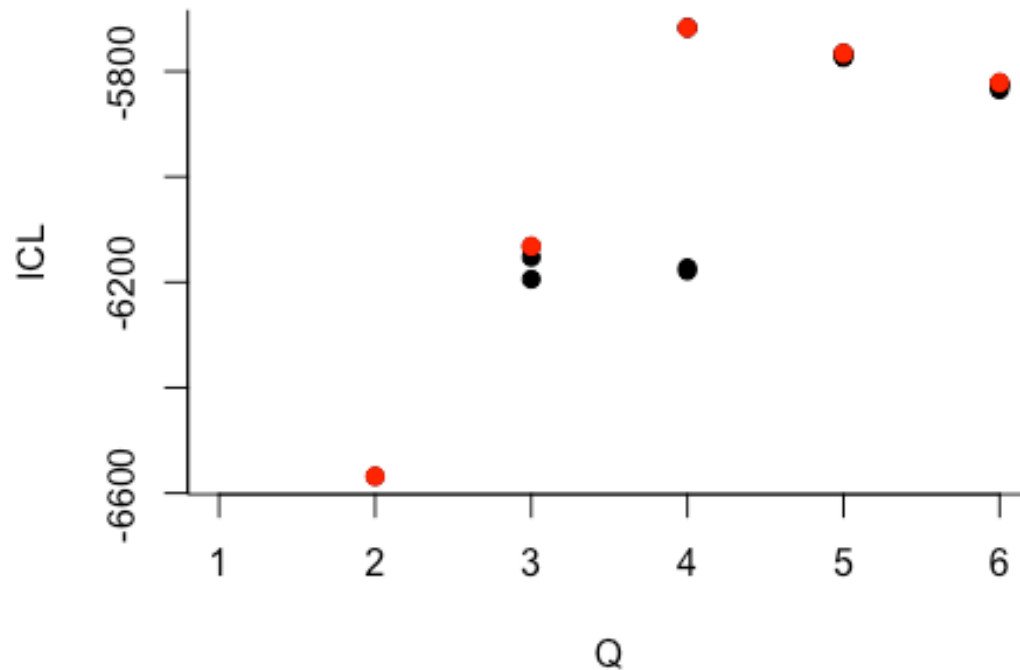
```
##      -> For 4 groups
##      -> Selecting initialization
##      -> Init from spectral clustering
##      -> Init from splitting groups from 3 groups
##      -> 4 initializations provided
##      -> 0 initializations already used
##      -> Estimation with 4 initializations
## Executing 4 jobs in parallel
Better ICL criterion found
##      -> new ICL: -5717.40060305038
##      -> old ICL: NA
```

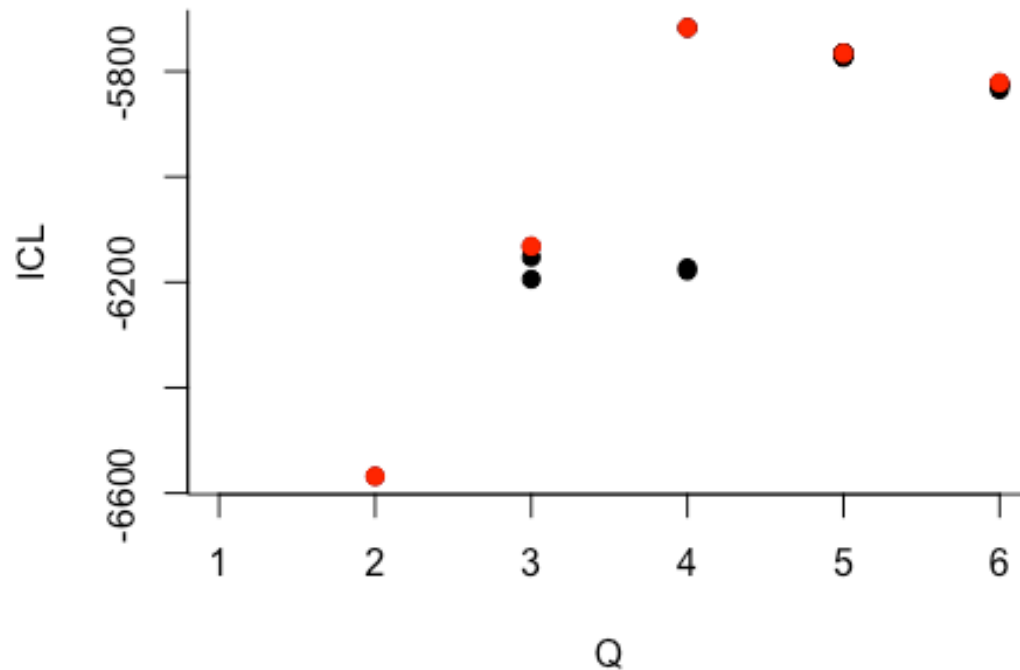
```
##      -> For 5 groups
##      -> Selecting initialization
##      -> Init from spectral clustering
##      -> Init from splitting groups from 4 groups
##      -> 5 initializations provided
##      -> 0 initializations already used
##      -> Estimation with 5 initializations
## Executing 5 jobs in parallel
Better ICL criterion found
##      -> new ICL: -5765.79717128072
##      -> old ICL: NA
```



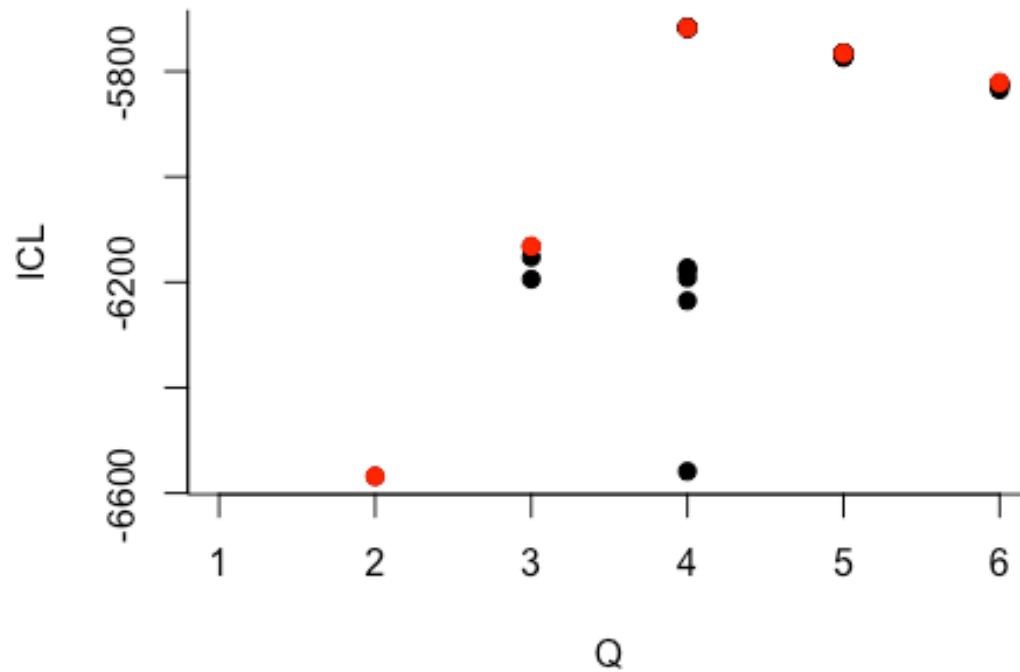
```
##      -> For 6 groups
##      -> Selecting initialization
##      -> Init from spectral clustering
##      -> Init from splitting groups from 5 groups
##      -> 6 initializations provided
##      -> 0 initializations already used
##      -> Estimation with 6 initializations
## Executing 6 jobs in parallel
Better ICL criterion found
##      -> new ICL: -5821.32844340961
##      -> old ICL: NA
```



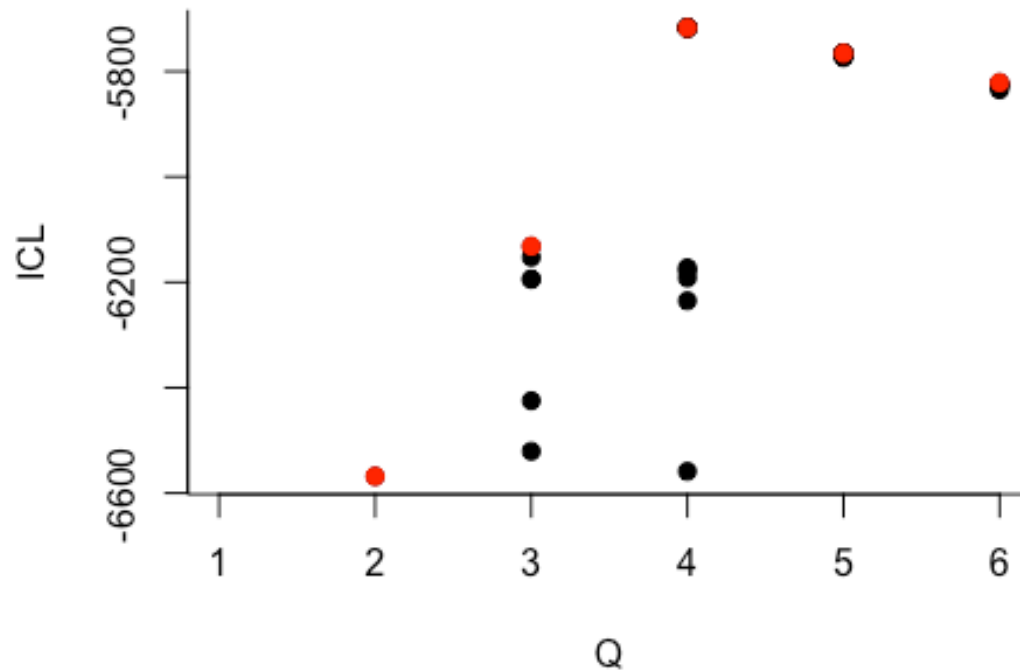
```
##      -> With descending number of groups
##      -> For 5 groups
##      -> Selecting intializations
##          -> Init from merging groups from 6 groups
##          -> 15 initializations provided
##          -> 0 initializations already used
##          -> Computing intializations quality
## Executing 15 jobs in parallel                                ->
Estimation with 9 initializations
## Executing 9 jobs in parallel                                ->
Better ICL criterion found
##          -> new ICL: -5765.79658158437
##          -> old ICL: -5765.79717128072
```



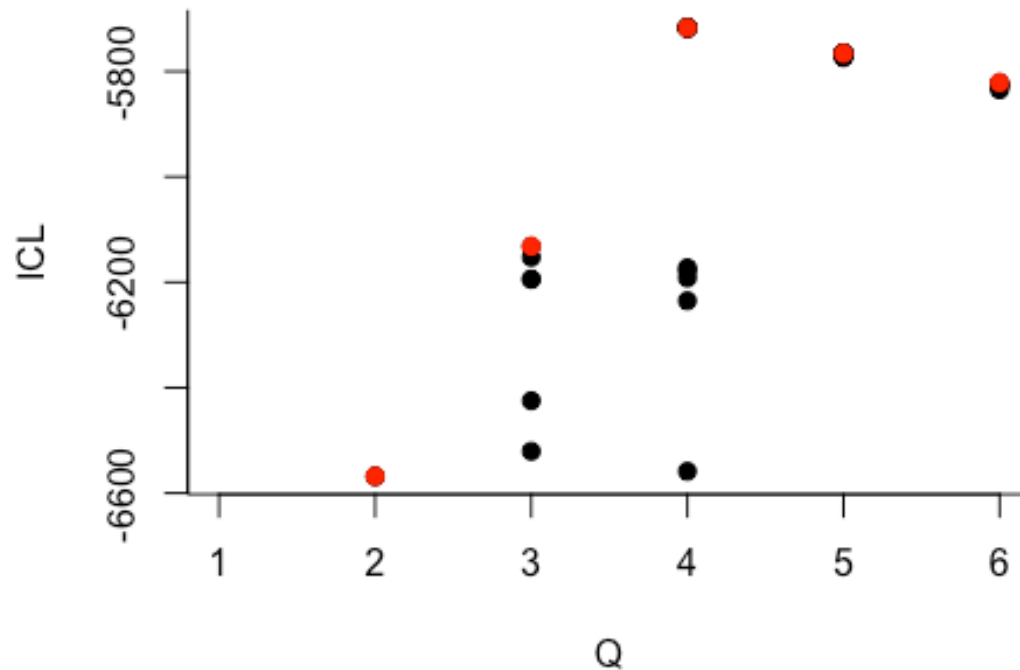
```
##      -> For 4 groups
##      -> Selecting intializations
##      -> Init from merging groups from 5 groups
##      -> 10 initializations provided
##      -> 1 initializations already used
##      -> Computing intializations quality
## Executing 10 jobs in parallel                                ->
Estimation with 8 initializations
## Executing 8 jobs in parallel                                ->
Useless, no better ICL criterion found
##      -> better ICL found: -5717.40060305038
##      -> old ICL: -5717.40060305038
```



```
##      -> For 3 groups
##      -> Selecting intializations
##      -> Init from merging groups from 4 groups
##      -> 6 initializations provided
##      -> 2 initializations already used
##      -> Estimation with 4 initializations
## Executing 4 jobs in parallel                                     ->
Useless, no better ICL criterion found
##      -> better ICL found: -6194.20087334228
##      -> old ICL: -6131.71308468876
```



```
##      -> For 2 groups
##      -> Selecting intializations
##      -> Init from merging groups from 3 groups
##      -> 3 initializations provided
##      -> 0 initializations already used
##      -> Estimation with 3 initializations
## Executing 3 jobs in parallel
Useless, no better ICL criterion found
##      -> better ICL found: -6568.13973745724
##      -> old ICL: -6568.13973745724
```



```
## -> Pass 2
##   -> With ascending number of groups
##     -> For 2 groups
##       -> Selecting initialization
##       -> Init from splitting groups from 1 groups
##       -> already done
##     -> For 3 groups
##       -> Selecting initialization
##       -> Init from splitting groups from 2 groups
##       -> already done
##     -> For 4 groups
##       -> Selecting initialization
##       -> Init from splitting groups from 3 groups
##       -> already done
##     -> For 5 groups
##       -> Selecting initialization
##       -> Init from splitting groups from 4 groups
##       -> already done
##     -> For 6 groups
##       -> Selecting initialization
##       -> Init from splitting groups from 5 groups
##       -> already done
##   -> With descending number of groups
```

```

##      -> For 5 groups
##      -> Selecting intializations
##      -> Init from merging groups from 6 groups
##      -> Already done
##      -> For 4 groups
##      -> Selecting intializations
##      -> Init from merging groups from 5 groups
##      -> Already done
##      -> For 3 groups
##      -> Selecting intializations
##      -> Init from merging groups from 4 groups
##      -> Already done
##      -> For 2 groups
##      -> Selecting intializations
##      -> Init from merging groups from 3 groups
##      -> Already done

sbm.mod$ICL

## [1] -7730.116 -6568.140 -6131.713 -5717.401 -5765.797 -5821.328

which.max(sbm.mod$ICL)

## [1] 4

```

Similarly to the Modularity based community detection, 4 clusters is optimal.

```

#model parameters
sbm.mod$model_parameters[[4]]

## $pi
##      [,1]      [,2]      [,3]      [,4]
## [1,] 0.461313128 0.06596443 0.063311999 0.01641017
## [2,] 0.061109928 0.45167548 0.012379386 0.07311981
## [3,] 0.085545750 0.01472271 0.445128771 0.01316306
## [4,] 0.009083286 0.07484070 0.009495395 0.45778834
##
## $n_parameters
## [1] 16

#membership probabilities
sbm.mod$memberships[[4]]$Z

##      [,1]      [,2]      [,3]      [,4]
## [1,] 0.000723589 0.000723589 0.000723589 0.997829233
## [2,] 0.000723589 0.000723589 0.000723589 0.997829233
## [3,] 0.000723589 0.000723589 0.000723589 0.997829233
## [4,] 0.000723589 0.000723589 0.000723589 0.997829233
## [5,] 0.000723589 0.000723589 0.000723589 0.997829233
## [6,] 0.000723589 0.000723589 0.000723589 0.997829233
## [7,] 0.000723589 0.000723589 0.000723589 0.997829233
## [8,] 0.000723589 0.000723589 0.000723589 0.997829233

```


[illegible]

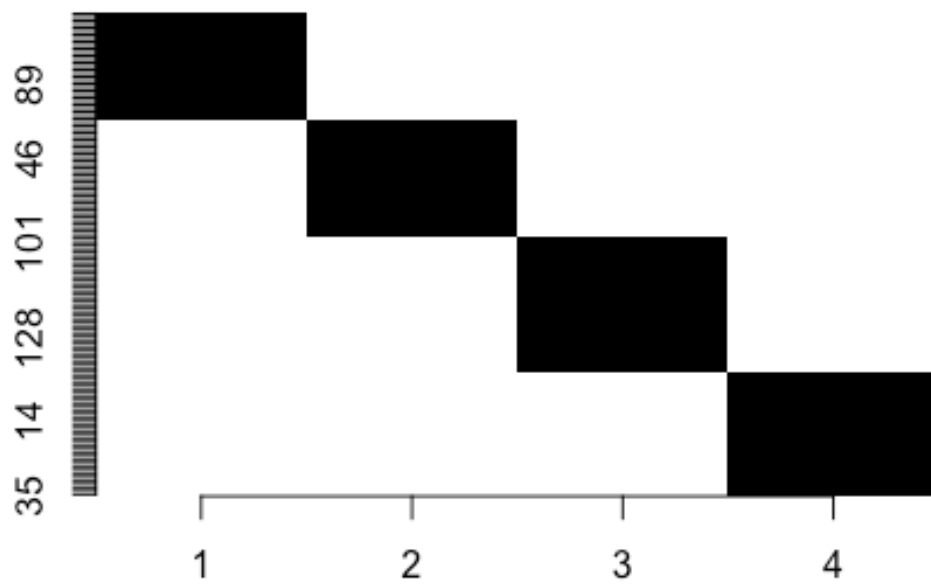
[illegible]

#which class deos each chimpanzee belong to.

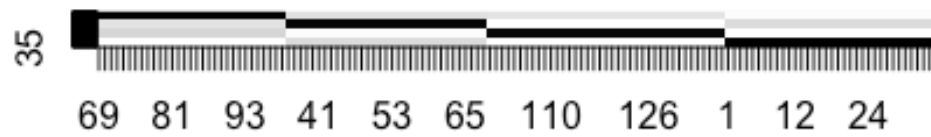
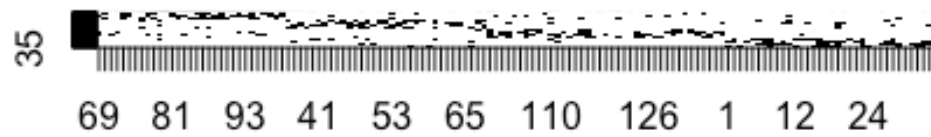
```
##      [1] 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4  
4 2 2  
##    [38] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1  
1 1 1  
##    [75] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 3 3 3 3 3 3 3 3 3  
3 3 3  
##   [112] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
```

```
#plot membership probabilities
```

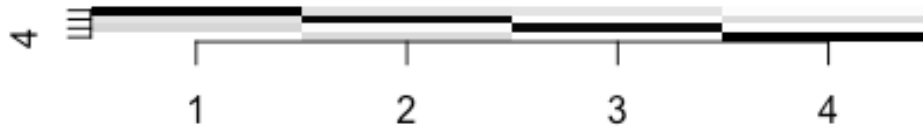
```
sbm.mod$memberships[[4]]$plot()
```



```
#observed ties Vs predicted ties  
sbm.mod$plot_obs_pred(4)
```



```
sbm.mod$plot_parameters(4)
```



The membership plot gives us 4 equal classes with equal probabilities for each class.

The next plot shows the actual connections. The third plot shows probability of connection. As we can see, the probabilities are very equal suggesting that the model fits the data well. The model parameters plot also shows how the chimpanzees are spread evenly across the 4 classes.

Evaluating the class memberships distribution:

```
class.member<-apply(sbm.mod$memberships[[4]]$Z,1,which.max)
table(class.member)

## class.member
##  1  2  3  4
## 31 33 39 35

# Percentage in each class
prop.table(table(class.member))

## class.member
##          1          2          3          4
## 0.2246377 0.2391304 0.2826087 0.2536232
```

The proportion of chimpanzees in each class is approximately equal. The plots produced similarly demonstrated that the chimpanzees are spread equally between all classes.

We will now assess how the groups created compare with the covariates. First, we will attach the class member to the data set.

```
data2<- cbind(attributes, class.member)
group1<- data2%>%filter(class.member==1)
group2<-data2%>%filter(class.member==2)
group3<-data2%>%filter(class.member==3)
group4<-data2%>%filter(class.member==4)
table(group1$Age, group1$Gender)

##
##           Female
## middle-aged    31

table(group2$Age, group2$Gender)

##
##           Male
## middle-aged    33

table(group3$Age, group3$Gender)

##
##           Female
## elderly        39

table(group4$Age, group4$Gender)

##
##           Male
## young          35

table(group1$SleepLoc)

##
## Loc1 Loc2
##   18   13

table(group2$SleepLoc)

##
## Loc1 Loc2
##   17   16

table(group3$SleepLoc)

##
## Loc1 Loc2
##   16   23
```

```
table(group4$SleepLoc)
```

```
##  
## Loc1 Loc2  
## 18 17
```

The first group is entirely female and middle aged the second group is entirely male and middle aged the third group is entirely female and elderly the fourth group is entirely male and young.

The sleeping location has no affect on the groups.

All this information matched exactly in line with what we disused throughout this analysis. Chimpanzees from the same gender and age will be more likely to stick with each other. However, the preferred sleeping location had no affect of the grooming of the chimpanzees.

We will now compare the community structures of the Modularity based community detection and the Stochastic block model (SBM).

```
table(cs$membership , class.member)
```

```
##      class.member  
##      1  2  3  4  
## 1  0  0  0 35  
## 2  0 33  0  0  
## 3 31  0  0  0  
## 4  0  0 39  0
```

This table shows the number of chimpanzees who allocated in each class for both the methods. As we cal see, all 35 chimpanzees who were allocated to be in class 1 for Modularity were in class 4 for SBM. It is clear that both methods allocated the groups in exactly the same way, only the title of the groups were different.

Now we can compute similarity measures:

```
chisq.test(table(cs$membership , class.member))
```

```
##  
## Pearson's Chi-squared test  
##  
## data:  table(cs$membership, class.member)  
## X-squared = 414, df = 9, p-value < 2.2e-16
```

The P value is significant.

Next, we can explore the Adjusted Rand method:

```
print(c('Adjusted  
Rand',compare(cs$membership,class.member,method="adjusted.rand")))  
## [1] "Adjusted Rand" "1"
```


As this result is 1 (as expected), we can conclude that the results from the two statistics are identical.

As the two results are identical, we know that this is the best way to cluster these chimpanzees.